

**Search Forms**  
**Search Results**

**Help**  
**User Searches**  
**Preferences**  
**Logout**

## Refine Search

### Search Results -

Terms	Documents
L1 and tax near (refund or estimate)	7

**Database:**  
US Pre-Grant Publication Full-Text Database  
US Patents Full-Text Database  
US OCR Full-Text Database  
EPO Abstracts Database  
JPO Abstracts Database  
Derwent World Patents Index  
IBM Technical Disclosure Bulletins

**Search:**    **Refine Search**

**Recall Text**  **Clear** **Interrupt**

---

### Search History

---

**DATE:** Wednesday, January 26, 2005 [Printable Copy](#) [Create Case](#)

<u>Set Name</u>	<u>Query</u>	<u>Hit Count</u>	<u>Set Name</u>
side by side			result set
<i>DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR</i>			
<u>L2</u>	L1 and tax near (refund or estimate)	7	<u>L2</u>
<u>L1</u>	705/19	106	<u>L1</u>

END OF SEARCH HISTORY

# Freeform Search

---

**Database:**

- US Pre-Grant Publication Full-Text Database
- US Patents Full-Text Database
- US OCR Full-Text Database
- EPO Abstracts Database
- JPO Abstracts Database
- Derwent World Patents Index
- IBM Technical Disclosure Bulletins

**Term:**

**Display:**  Documents in **Display Format:**  Starting with Number **Generate:**  Hit List  Hit Count  Side by Side  Image

---

---

## Search History

---

**DATE:** Wednesday, January 26, 2005 [Printable Copy](#) [Create Case](#)

<u>Set Name</u>	<u>Query</u>	<u>Hit Count</u>	<u>Set Name</u>
		result set	
<i>DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR</i>			
<u>L16</u>	L15 and loan	64	<u>L16</u>
<u>L15</u>	l9 and (comput\$ or estimat\$)	181	<u>L15</u>
<u>L14</u>	(comput\$ or estimat\$) near tax near refund	0	<u>L14</u>
<u>L13</u>	L12 and estimat\$	22	<u>L13</u>
<u>L12</u>	l9 and loan	71	<u>L12</u>
<u>L11</u>	L10 and loan	22	<u>L11</u>
<u>L10</u>	L9 and estimat\$	46	<u>L10</u>
<u>L9</u>	tax with refund	223	<u>L9</u>
<u>L8</u>	estimat\$ near tax with refund	0	<u>L8</u>
<u>L7</u>	estimat\$ near tax near refund	0	<u>L7</u>
<u>L6</u>	L4 and refund	0	<u>L6</u>
<u>L5</u>	L4 snd refund	21443	<u>L5</u>
<u>L4</u>	estimat\$ near tax near payment	5	<u>L4</u>
<u>L3</u>	L2 and estimat\$	0	<u>L3</u>
<u>L2</u>	(annual or yearly) near tax with refund	1	<u>L2</u>
<u>L1</u>	estimat\$ near (annual or yearly) near tax with refund	0	<u>L1</u>

END OF SEARCH HISTORY

## Freeform Search

**Database:** US Pre-Grant Publication Full-Text Database  
US Patents Full-Text Database  
US OCR Full-Text Database  
EPO Abstracts Database  
JPO Abstracts Database  
Derwent World Patents Index  
IBM Technical Disclosure Bulletins

**Term:**



**Display:**  Documents in Display Format:  Starting with Number

**Generate:**  Hit List  Hit Count  Side by Side  Image

---

### Search History

---

**DATE:** Wednesday, January 26, 2005 [Printable Copy](#) [Create Case](#)

<u>Set Name</u>	<u>Query</u>	<u>Hit Count</u>	<u>Set Name</u>
side by side			result set
	DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR		
<u>L12</u>	L11 and (estimate or calculate or compute)	18	<u>L12</u>
<u>L11</u>	income near tax near refund	43	<u>L11</u>
<u>L10</u>	income near tax	974	<u>L10</u>
<u>L9</u>	L8 and refund	30	<u>L9</u>
<u>L8</u>	comput\$ near tax	338	<u>L8</u>
<u>L7</u>	(pre-comput\$ or precomput\$) near tax	0	<u>L7</u>
<u>L6</u>	pre-comput\$ near tax	0	<u>L6</u>
<u>L5</u>	pre-comput\$ near tax near refund	0	<u>L5</u>
<u>L4</u>	L3 and loan	11	<u>L4</u>
<u>L3</u>	L2 and estimat\$	17	<u>L3</u>
<u>L2</u>	L1 and refund	57	<u>L2</u>
<u>L1</u>	tax near account\$	538	<u>L1</u>

END OF SEARCH HISTORY

**Search Forms**  
**Search Results**

**Help****User Searches**

US Pre-Grant Publication Full-Text Database

**Preferences**

US Patents Full-Text Database

**Logout**

US OCR Full-Text Database

**Database:**

EPO Abstracts Database

JPO Abstracts Database

Derwent World Patents Index

IBM Technical Disclosure Bulletins

**Term:**

5138549.uref.



**Display:**  **Documents in Display Format:**  **Starting with Number:**

**Generate:**  **Hit List**  **Hit Count**  **Side by Side**  **Image**

**Search** **Clear** **Interrupt**

### Search History

**DATE:** Wednesday, January 26, 2005 [Printable Copy](#) [Create Case](#)

**Set Name** **Query**  
side by side

**Hit Count** **Set Name**  
result set

*DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR*

<u>L45</u>	5193057.uref.	14	<u>L45</u>
<u>L44</u>	5138549.uref.	15	<u>L44</u>
<u>L43</u>	4970655.uref.	78	<u>L43</u>
<u>L42</u>	6202052.uref.	3	<u>L42</u>

*DB=USPT; PLUR=YES; OP=OR*

<u>L41</u>	4970655.pn.	1	<u>L41</u>
------------	-------------	---	------------

*DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR*

<u>L40</u>	6202052.pn.	2	<u>L40</u>
<u>L39</u>	5138549.pn.	2	<u>L39</u>
<u>L38</u>	5193057.pn.	2	<u>L38</u>

*DB=USPT; PLUR=YES; OP=OR*

<u>L37</u>	US-5963921-A.did.	1	<u>L37</u>
<u>L36</u>	US-5963921-A.did.	1	<u>L36</u>
<u>L35</u>	4971363.pn.	1	<u>L35</u>
<u>L34</u>	5114009.pn.	1	<u>L34</u>
<u>L33</u>	5188562.pn.	1	<u>L33</u>

<u>L32</u>	5277451.pn.	1	<u>L32</u>
<u>L31</u>	4713761.pn.	1	<u>L31</u>
<u>L30</u>	4727243.pn.	1	<u>L30</u>
<u>L29</u>	5287268.pn.	1	<u>L29</u>
<u>L28</u>	5521815.pn.	1	<u>L28</u>
<u>L27</u>	5555497.pn.	1	<u>L27</u>
<u>L26</u>	5694322.pn.	1	<u>L26</u>
<u>L25</u>	4321672.pn.	1	<u>L25</u>
<u>L24</u>	4321672.pn.	1	<u>L24</u>
<u>L23</u>	4597046.pn.	1	<u>L23</u>
<u>L22</u>	4648037.pn.	1	<u>L22</u>
<u>L21</u>	4694397.pn.	1	<u>L21</u>
<u>L20</u>	4736294.pn.	1	<u>L20</u>
<u>L19</u>	5623403.pn.	1	<u>L19</u>
<u>L18</u>	5774872.pn.	1	<u>L18</u>
<u>L17</u>	5774872.pn.	1	<u>L17</u>
<i>DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR</i>			
<u>L16</u>	I4 and 705/31	9	<u>L16</u>
<i>DB=USPT; PLUR=YES; OP=OR</i>			
<u>L15</u>	4876648.pn.	1	<u>L15</u>
<u>L14</u>	5724523.pn.	1	<u>L14</u>
<u>L13</u>	5724523.pn.	1	<u>L13</u>
<i>DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR</i>			
<u>L12</u>	L10 and (estimat\$ near tax or estimat\$ with tax)	4	<u>L12</u>
<u>L11</u>	L10 and estimat\$ near refund	0	<u>L11</u>
<u>L10</u>	(income with tax with refund or income near tax near refund)	57	<u>L10</u>
<i>DB=USPT; PLUR=YES; OP=OR</i>			
<u>L9</u>	5832461.pn.	1	<u>L9</u>
<u>L8</u>	5864828.pn.	1	<u>L8</u>
<u>L7</u>	US-5946668-A.did.	1	<u>L7</u>
<i>DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR</i>			
<u>L6</u>	L4 and estimat\$	7	<u>L6</u>
<u>L5</u>	L4 and estimat\$ near tax near refund	0	<u>L5</u>
<u>L4</u>	income near tax near refund	43	<u>L4</u>
<u>L3</u>	L1 and estimat\$ near2 income near2 tax	16	<u>L3</u>
<u>L2</u>	L1 and estimat\$ adj2 tax	42	<u>L2</u>
<u>L1</u>	705.clas.	31452	<u>L1</u>

END OF SEARCH HISTORY

## Refine Search

## Search Results -

<b>Terms</b>	<b>Documents</b>
(tax with refund or tax near refund)	223

**Database:** US CCR Full Text Database  
EPO Abstracts Database  
JPO Abstracts Database  
Derwent World Patents Index  
IBM Technical Disclosure Bulletins

### Search:

A small rectangular icon with a vertical striped pattern and a magnifying glass symbol inside.

Refine Search

**Recall Text** 

**Clear**

## Interrupt

## Search History

**DATE:** Wednesday, January 26, 2005 [Printable Copy](#) [Create Case](#)

## Set Name Query

## **side by side**

### Hit Count Set Name

### **result set**

*DB=PGPB.USPT.USOC.EPAB.JPAB.DWPI.TDBD; PLUR=YES; OP=OR*

<u>L12</u>	(tax with refund or tax near refund)	223	<u>L12</u>
<u>L11</u>	L10 and refund	16	<u>L11</u>
<u>L10</u>	tax near preparer	44	<u>L10</u>
<u>L9</u>	"jackson hewitt".as.	0	<u>L9</u>
<u>L8</u>	"h and r block"	0	<u>L8</u>
<u>L7</u>	"jackson hewitt"	0	<u>L7</u>
<u>L6</u>	L5 and history	41	<u>L6</u>
<u>L5</u>	L4 and refund	125	<u>L5</u>
<u>L4</u>	(income with tax or income near tax)	1367	<u>L4</u>
<u>L3</u>	L1 and (refund or repayment or re-payment)	1	<u>L3</u>
<u>L2</u>	L1 and refund	1	<u>L2</u>
<u>L1</u>	(income with tax with history or income near tax near history)	20	<u>L1</u>

END OF SEARCH HISTORY

# Freeform Search

---

US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
<b>Database:</b> EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Term:

Display:  10 Documents in Display Format:  Starting with Number  1

Generate:  Hit List  Hit Count  Side by Side  Image

---

---

## Search History

---

DATE: Wednesday, January 26, 2005 [Printable Copy](#) [Create Case](#)

Set Name Query

side by side

Hit Count Set Name  
result set

<i>DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR</i>			
<u>L12</u>	(carryback or pre-tax) same (refund or adjustment or loan)	34	<u>L12</u>
<u>L11</u>	(carryback or pre-tax) same (refund or adjustment)	17	<u>L11</u>
<u>L10</u>	(carryback or pre-tax) same refund	2	<u>L10</u>
<u>L9</u>	L8 and tax near2 refund	4	<u>L9</u>
<u>L8</u>	carry same back	235623	<u>L8</u>
<u>L7</u>	L5 and carryback	0	<u>L7</u>
<u>L6</u>	L5 and (carryback or carry near back)	0	<u>L6</u>
<u>L5</u>	tax near refund	141	<u>L5</u>
<u>L4</u>	carryback same tax same refund	0	<u>L4</u>
<u>L3</u>	carryback with tax with refund	0	<u>L3</u>
<u>L2</u>	carryback near tax near refund	0	<u>L2</u>
<u>L1</u>	carryback with tax refund	3646	<u>L1</u>

END OF SEARCH HISTORY

## Refine Search

---

### Search Results -

Terms	Documents
pre-tax with refund	1

---

**Database:**

- US Pre-Grant Publication Full-Text Database
- US Patents Full-Text Database
- US OCR Full-Text Database
- EPO Abstracts Database
- JPO Abstracts Database
- Derwent World Patents Index
- IBM Technical Disclosure Bulletins

**Search:**

Refine Search

Recall Text
Clear
Interrupt

---

### Search History

---

**DATE:** Wednesday, January 26, 2005    [Printable Copy](#)    [Create Case](#)

<u>Set</u>	<u>Name</u>	<u>Query</u>	<u>Hit Count</u>	<u>Set Name</u>
side by side				result set
		DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR		
<u>L22</u>		pre-tax with refund	1	<u>L22</u>
<u>L21</u>		pre-tax near (refund or loan)	0	<u>L21</u>
<u>L20</u>		advanc\$ near tax near refund	0	<u>L20</u>
<u>L19</u>		advanced near tax near refund	0	<u>L19</u>
<u>L18</u>		automated near tax near refund	0	<u>L18</u>
		DB=USPT; PLUR=YES; OP=OR		
<u>L17</u>		4554418.pn.	1	<u>L17</u>
<u>L16</u>		4554418.pn.	1	<u>L16</u>
		DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR		
<u>L15</u>		(online or on-line or internet) same tax with refund	11	<u>L15</u>
<u>L14</u>		L12 and (prior or advance) near data	12	<u>L14</u>
<u>L13</u>		L12 and (prior or advance) near data	0	<u>L13</u>
<u>L12</u>		L11 and estimat\$	209	<u>L12</u>

<u>L11</u>	(tax near refund or tax near return)	712	<u>L11</u>
<u>L10</u>	(tax near refund near estimat\$)	0	<u>L10</u>
<u>L9</u>	estimat\$ near tax near refund	0	<u>L9</u>
<u>L8</u>	(proforma with tax with return or pro-forma with tax with return or pro-forma near tax near return or proforma near tax near return)	0	<u>L8</u>
<u>L7</u>	l4 and (loan or cash near advance)	60	<u>L7</u>
<u>L6</u>	l4 and estimat\$ near return	2	<u>L6</u>
<u>L5</u>	L4 and estimat\$	31	<u>L5</u>
<u>L4</u>	tax near refund	141	<u>L4</u>
<u>L3</u>	(proforma near tax near return or estimat\$ near tax near refund)	0	<u>L3</u>
<u>L2</u>	(proforma near tax near return or estimat\$ near tax near refund or estimat\$ near tax near return)	1	<u>L2</u>
<u>L1</u>	(proforma near tax near return or estimat\$ near liability or estimat\$ near tax near return)	82	<u>L1</u>

END OF SEARCH HISTORY

Membership Publications/Services Standards Conferences Careers/Jobs



RELEASE 1.8

Welcome  
United States Patent and Trademark Office



Help FAQ Terms IEEE Peer Review

Quick Links

Welcome to IEEE Xplore®

- Home
- What Can I Access?
- Log-out

#### Tables of Contents

- Journals & Magazines
- Conference Proceedings
- Standards

#### Search

- By Author
- Basic
- Advanced
- CrossRef

#### Member Services

- Join IEEE
- Establish IEEE Web Account
- Access the IEEE Member Digital Library

#### IEEE Enterprise

- Access the IEEE Enterprise File Cabinet

Print Format

[Search Results](#) [PDF FULL-TEXT 436 KB] [DOWNLOAD CITATION](#)



## The Carryback Processing Assistant: a study in case processing

Schreiber, R.K. Patil, R.

Internal Revenue Service, Washington, DC, USA;

*This paper appears in: AI Systems in Government Conference, 1989., Proc of the Annual*

Meeting Date: 03/27/1989 - 03/31/1989

Publication Date: 27-31 March 1989

Location: Washington, DC USA

On page(s): 218 - 225

Reference Cited: 6

Inspec Accession Number: 3428674

#### Abstract:

Case processing involves amending, correcting, or updating an existing account least automated type of processing in the Internal Revenue Service. The author describes how a constraint network and a process called discrepancy resolution used to limit the amount of input data needed to process a case. He presents re achieved by the Carryback Processing Assistant, a system designed to partially the processing of the carryback claims received by an IRS Service Center

#### Index Terms:

[expert systems](#) [financial data processing](#) [government data processing](#) [Carryback Process Assistant](#) [Internal Revenue Service](#) [carryback claims](#) [case processing](#) [constraint netw](#) [discrepancy resolution](#) [tax refunds](#)

#### Documents that cite this document

There are no citing documents available in IEEE Xplore at this time.

[Search Results](#) [PDF FULL-TEXT 436 KB] [DOWNLOAD CITATION](#)

# THE CARRYBACK PROCESSING ASSISTANT: A STUDY IN CASE PROCESSING

by

Richard K. Schreiber, Internal Revenue Service  
Ramesh Patil, Massachusetts Institute of Technology

## Abstract

Case processing involves amending, correcting, or updating an existing account. It is the least automated type of processing in the Internal Revenue Service. This article describes how a constraint network and a process called "discrepancy resolution" can be used to limit the amount of input data needed to process a case. We present results achieved by the Carryback Processing Assistant, a system designed to partially automate the processing of the carryback claims received by an IRS Service Center.

## A Characterization of Case Processing

We have analyzed the work of an IRS Service Center into the categories of case processing and input processing. A large workforce in the ten IRS processing centers is devoted to each type of task. Input processing centers around the entry of data into the computer system and is characterized by task specialization or horizontal integration. An example is the initial processing of tax returns. Case processing on the other hand involves amending, correcting, or updating an existing case or account. Making an adjustment to a tax account in response to a taxpayer request is an example of case processing.

The difference between case processing and input processing lies in the amount of data to be entered, the complexity of the analysis to be performed, and whether entered data will be saved for future use. For case processing, the source document and mixed media documents such as electronic transcripts and returns from

the file are assembled and the case is worked at a single point in time. For input processing each step is separated in time which results in additional steps to retrieve documents and make corrections. These differences may be clearly seen in Figure 1.

Almost no case processing tasks have been automated. Normal automation of a case processing task would involve converting it into an input processing task. However, the relatively large amount of data needed to resolve a case and the level of expertise required to perform analysis of the case have proved in the past to be effective barriers against automation.

The rapid growth of expert systems technology has changed this equation by providing the tools and techniques to capture the levels of expertise needed for case processing. Goal directed questioning can limit the amount of input data needed to resolve a case, but questions must greatly restrict the amount of data needed in order to overcome the inherent slowness of the Question & Answer technique. In the domain of tax processing the expense of obtaining input data continues to loom as a formidable barrier.

In the following sections we will present our solution to the problem of case processing. We will describe a program designed to assist in the processing of one type of case, tentative carrybacks. After examining the components of this system, we will describe the results it achieved in a field test, and conclude by describing why our techniques may apply to many case processing domains.

<u>Case Processing</u>	<u>Input Processing</u>
Data enter input document.	
<b>Subsequent Processing</b> (Error Correction Loop)	
Format entered data into transaction to post to file.	Analyze input document & identify inconsistencies.
Analyze input document & correct inconsistencies.	
Retrieve document & correct inconsistencies.	
<b>Document Posting</b> (Unpostable Loop)	
Analyze file and correct inconsistencies with input document(s).	Analyze file and identify inconsistencies with input document(s).
Retrieve document & correct inconsistencies (Unpostables).	
Generate transaction to post to file.	Post corrections to file.
Figure 1. List of tasks associated with case processing input processing. In this model the tasks differ because data entry, analysis of the source document, and analysis of the file occur at different times for input processing but occur simultaneously for case processing.	

#### Design For a Case Processing Assistant

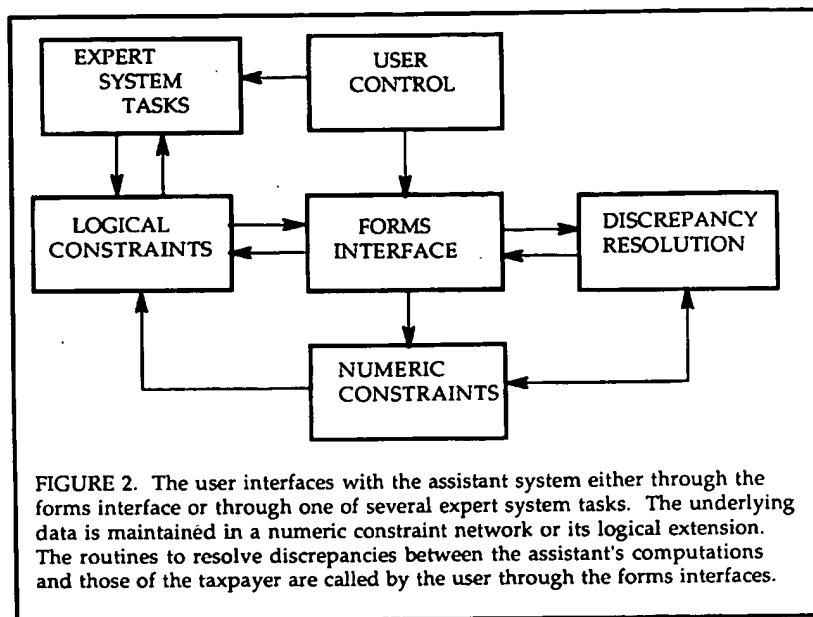
The primary determinant of the shape of case processing is the lack of input data and the expense of acquiring it. A basic strategy is to enter a small subset of the data and use it to generate an approximation of the complete data set. This approach seems to be particularly powerful in the IRS forms domain.

Once the minimum data set has been entered the technician must define goals to the assistant either explicitly or implicitly. These goals form the basis for testing the results generated by the assistant and should cause it to repair its own calculations using supplied target figures. In a more elaborate technique the goals themselves can be called into question when the system is unable to find a causal basis for supporting them.

Expert systems have proven that they can deliver expert analysis and can pursue an

intelligent strategy for acquiring data. However, unless the system is provided with a complete set of input data it cannot be guaranteed to reach a satisfactory conclusion. This indeterminacy suggests that some explicit control or directions from the user will be necessary and for this reason we have looked for a partial solution in the form of an expert assistant.

The assistant paradigm is well established and we will only briefly comment on it. An assistant provides an intelligent environment. It makes assumptions, performs certain types of operations automatically, provides a number of computational tools which may be used by the technician or at the direction of the technician. The assistant is capable of taking actions or proposing actions for review by the technician. While the assistant is not capable of independently processing a case, it works interactively with the technician toward that goal.



#### The Carryback Domain

A carryback is an income tax claim (filed on an Amended Return, Form 1040X) or a tentative application (filed on Form 1045). A carryback is generated by negative income -- called a net operating loss -- or by tax credits that exceed income in a given year. The excess loss or credit is then "carried back" to a previous year. A loss is used to reduce adjusted gross income in the previous year, while a credit is applied directly against tax. The result in both cases is a tax refund.

The procedures governing the computation and application of net operating loss are complex. So much so, that one study showed that forty-nine per cent of the carryback cases contain errors. It takes approximately one year of training and experience to produce a knowledgeable tax examiner. It takes another year of training and specialization to produce a knowledgeable carryback examiner.

The basic processing scenario for working a case is as follows:

Received cases are preprocessed to obtain printed account transcripts and to review

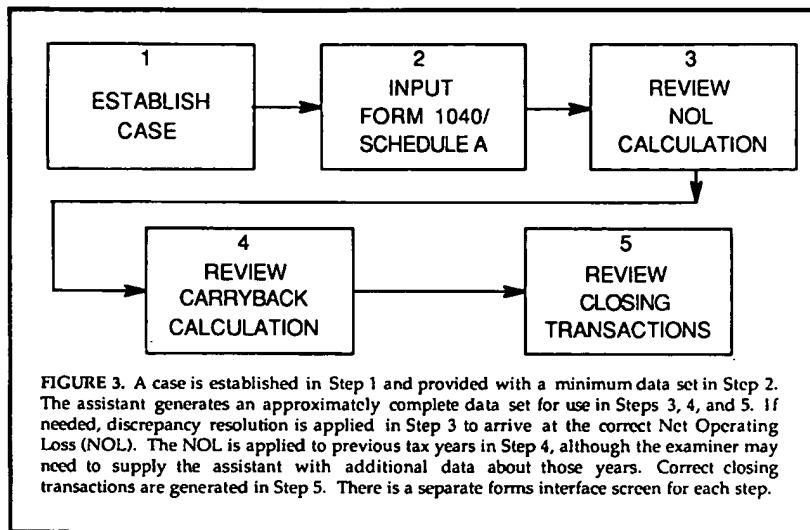
some related issues such as timeliness and completeness of the case.

Processable cases are assigned to a tax examiner who analyzes the case, verifies the computations, and determines whether additional case documents need to be retrieved from the file or whether processing on the case needs to be coordinated with any other units.

When analysis of the case is completed, closing transactions are input to adjust the tax, modify the data base account and issue a refund to the taxpayer.

#### System Components

A user interacts with the Carryback Assistant and directs processing through the forms interface which sits on top of a logical and numeric constraint system. The constraint system generates the values for several of the forms. Discrepancy resolution or debugging can be initiated by the user or by the constraint system. An expert task system uses the logical constraint system, but interacts directly with the user. A schematic of the Carryback Processing Assistant is given in Figure 2.



**Forms Interface.** The Carryback Assistant is implemented as a series of forms associated with the case. This includes the Individual Income Tax Form, Form 1040 (pages one and two); the Schedule of Deductions, Schedule A; the Net Operating Loss Computation, Form 1045 - Schedule A; the Application of Carryback Credits, Form 1045; and other supporting forms such as the Alternative Minimum Tax Computation, Form 6251.

An interesting issue in designing screen forms is deciding the appropriate degree of forms mimicry. If we choose to exactly represent each form, a complete form cannot be displayed on a single screen. At every decision point where there is a conflict between exactly representing a form and making computations explicitly available to the user or providing a concise display, we have followed the most functional line.

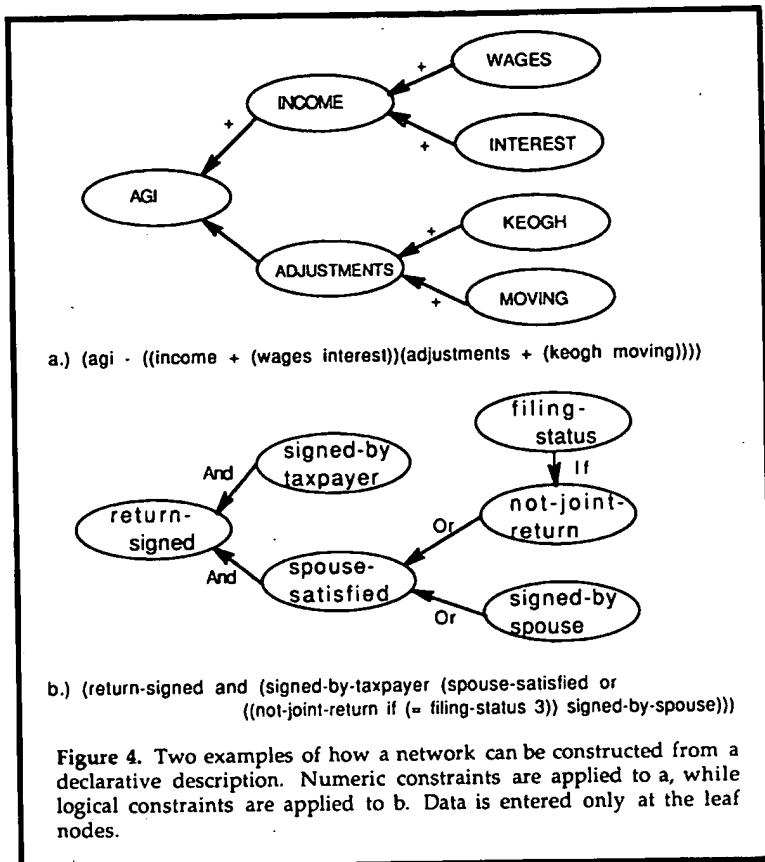
Our approach in designing forms has been to eliminate scrolling, to combine displays, to provide redundant displays, and to describe computations as fully as possible. We have eliminated scrolling by putting each form on a separate screen. This requires a more abstract form and detracts from the visual authenticity of the displayed form, but it accelerates processing and in practice we have found it to

communicate all necessary information to the user.

At the same time, we have added additional information to several forms in order to show explicitly how the assistant arrived at a particular total. While an explanation for a given computation may be requested, wherever possible we have tried to display all items entering into a computation.

The user controls the system by moving from screen to screen. A common sequence is shown in Figure 3. Constraints are applied only when the local network for a particular screen is activated. Debugging is also initiated when the user identifies a discrepancy between the assistant's and the taxpayer's computations. The user is also provided with the ability to override any of the assistant's computations or conclusions.

**Numeric Constraints.** All variables are linked using a constraint system we developed for this project. The constraint system acts as an automatic programmer in converting a declarative description into rules which are fired whenever the value of a variable changes. The advantage of this constraint system is that building a network from a declarative description is easy and the network can easily



**Figure 4.** Two examples of how a network can be constructed from a declarative description. Numeric constraints are applied to a, while logical constraints are applied to b. Data is entered only at the leaf nodes.

be changed. Figure 4 gives an example of a small network and the declarative description from which it is constructed. A disadvantage of constraints is that quiescence must be achieved after each input item is entered, and that in very large networks reaching quiescence may be time consuming. To avoid the latter problem we have partitioned our network into several smaller networks.

The constraint system may also serve other purposes. For example, we also use it to control the graphic redisplay of changing values and the computation of complementary values. This is achieved by independently attaching rules or operators to nodes. We perform this when the network is initialized.

Logical Constraints. We have extended the constraint system discussed above to process the logical operators "and," "or," "not," "if," and "then". "If" is the main device for linking numerically constrained values with logically constrained values, while "then" links the logical network to the numeric network. A sample logical network is shown in Figure 4.

The constraint logic is four-valued with the values of "True", "False", "Undetermined", and "Unknown". "Unknown" means that the user has been queried about the value of a particular variable, but has been unable to supply a value. "Undetermined" means that the user has not been queried.

In practice, "unknowns" usually occur when the taxpayer has not supplied one or more items of required data. Rather than rejecting the case, the examiner will process the case if the "required" data item would not change the net processing result. If a leaf node is set to "unknown", conjunctive clauses cause the value "unknown" to be propagated forward. Only disjunctive clauses offer the possibility of leading to a conclusion other than "unknown".

Forward chaining is accomplished by the normal operation of the constraint system, while backward chaining is defined as a separate function. Backward chaining is initiated when the user wishes to activate an expert system task. The backward chainer searches for leaf nodes with the value of "undetermined" and triggers questions to the user. Although the backward chainer can be applied to any subgoal, can be exited and re-entered at any point in the processing, or can be terminated as soon as a conclusion other than "undetermined" is reached, it is generally useful in our domain to stop processing if the conclusion is "true" and to continue processing if the conclusion is "false" in order to identify all conditions that would have to be corrected to reach a "true" conclusion.

Discrepancy Resolution. Reasoning about causal information, measuring it against some target value(s), and applying debugging procedures to the causal links is well described by Reid Simmons as the generate, test, and debug paradigm. In the debugging paradigm the primitive nodes in a causal net are located, discrepancies are identified by regressing through the nodes, and the discrepancies are repaired by applying one of six domain independent techniques. Multiple discrepancies may be resolved with hill climbing.

Discrepancy resolution refers to resolving differences between values generated by the system and values supplied by the taxpayer. This process is close in spirit to generate, test, and debug; but computationally it is quite different. The major difference is that the goal or target figure is not necessarily correct. Either the system or the taxpayer may be correct, or both may be in error. Moreover, we

do not have access to each item in the taxpayer's computation -- normally, only a total value is provided. A partial solution is not helpful. If we cannot identify all of the causes for a discrepancy there is little reason to believe a partial solution should be preferred to the initial system assumptions.

Discrepancies arise because the system has made an incorrect assumption(s), or because it lacks input data. Correspondingly, the taxpayer may make errors in classifying income or deductions or by omitting totals that should be included in a computation. The taxpayer may also make computational errors. Occasionally, these may be true math errors, but they are more frequently caused by a misunderstanding of the various computation rules.

To resolve a discrepancy we identify the primitive nodes that could have contributed to the discrepancy and then perform a closure on all possible combinations of values for these nodes. A closure yields  $\sum_{k=1}^n \binom{n}{k}$  or  $2^n$  possibilities. Potentially, the number of combinations to be considered can be large -- the number of categories into which income from the Income Tax Return, Form 1040, can be analyzed is 28, so the number of combinations would exceed 200 million. In fact the average number of categories is more on the order of 28, and simple filters to eliminate impossible values routinely reduces the number to 25.

#### Current Results

In February and March of 1988 we tested the Carryback Processing Assistant at an IRS Service Center. Two hundred cases that had been previously processed and reviewed were selected for the test. Cases were screened to obtain an equal distribution over seven attributes we considered important to distinguish case types.

The Assistant processed seventy-five percent of the cases correctly. Fifty percent of the cases required discrepancy resolution, but the system was only able to resolve fifty percent of those discrepancies. Approximately two-thirds of the unresolved discrepancies could be

resolved by making some additional assumptions. But resolving the remaining discrepancies has caused us to design major revisions to the discrepancy resolution routines.

The most important technique in resolving discrepancies is the inductive closure we described above. Currently, if we are trying to resolve a nonbusiness income discrepancy, we compute a closure on the value of all primitive nodes above the node we are trying to resolve. Unfortunately, this computation does not consider any nodes that may have misanalyzed into another category (business). Nor is the analysis of other categories sufficient, since multiple misclassifications could have occurred. Multiple misclassifications can be accounted for by considering all types of income together and then testing to find permissible classifications.

A related step in the resolution process is to acquire additional information. We initially developed the logical constraint system and backward chainer in order to obtain additional information to aid in discrepancy resolution. Additional experience indicates this may not be necessary. We will either ask for specific data items or provide an interface that solicits the needed data.

In some cases, all efforts at resolution fail. If all pertinent data has been entered, it is likely that the goal or target amount is in error (the taxpayer's figure). The assistant can tentatively conclude that the taxpayer is in error, but it is more useful to present a display showing the assistant's computations and possible missing data items. This display serves as an index to the tax examiner in suggesting other possible resolutions. The examiner must decide whether to accept the assistant's figures or to override them.

An interesting result to us was the systems usefulness in detecting errors. We selected processed and reviewed cases so that we could be certain about the results and could identify errors made by the Carryback Processing Assistant. A surprising result, however, was that the system was able to identify or suggest errors made by tax examiners in 20 percent of the cases processed! This was a major success

for the assistant approach and established a solid basis for further development.

The user's ability to override the assistant's computations is a mixed blessing. We found several cases where the examiner incorrectly overrode the assistant's computations. We plan to permit the override, but incorporate additional displays to justify the assistant's computations and to solicit justification from the examiner.

We were unable to obtain usable production results because connection of the assistant to the existing data retrieval system could only be simulated. We expect a significant decrease in the time required to process a case once we can take advantage of a direct connection to the data base.

### Conclusion

There are significant barriers to applying expert systems technology to the domain of case processing. Although ES systems are well equipped to deliver the kind of expertise that is needed, they founder because of the need for input data (just as in the case of traditional systems) and because of the difficulty of establishing clear goals or targets.

We have successfully overcome both of these difficulties by using constraints to generate an approximately complete data set from a partial data set and by developing the technique of discrepancy resolution which allows debugging in the absence of a clear target. Although discrepancy resolution is potentially as capable of reaching a conclusion as a human expert, it is used primarily because it has access to less data than the human expert. Thus, it is most effectively implemented in the context of an interactive assistant.

These techniques can be widely applied to case processing problems in the tax processing domain. They may be applied to other case processing domains where the input consists of multiple interrelated or hierarchical forms, namely where data on one form is used to support computations on another form. These

techniques may also be applied in domains with relatively few variable values. In that case, default values can be propagated through the network, and discrepancy resolution can be used to identify the variable values, or to test different default values.

#### References

- [1] Laurel, Brenda K. "Interface as Mimesis," **User Centered System Design** edited by Donald A. Norman and Stephen W. Draper, Lawrence Erlbaum Associates 1986.
- [2] Rich, Charles and Waters, Richard C. "The Programmer's Apprentice: A Research Overview" in **Computer** magazine, Vol. 21, No. 11, November 1988.
- [3] Schreiber, Richard K. and Grady, Gerald J. "Expert System Applications in the IRS," **Journal of Policy Analysis & Management**, Volume 8, Number 2, 1989.
- [4] Simmons, Reid G., "A theory of Debugging Plans and Interpretations," **Proceedings of the Seventh National Conference on Artificial Intelligence**, Morgan Kaufmann 1988.
- [5] Steele, Guy L., "The Definition and Implementation of a Computer Programming Language Based on Constraints," **AI-TR-595**, M.I.T. 1980.
- [6] Waters, Richard C. "The Programmer's Apprentice: A Session with KBEmacs" **Artificial Intelligence and Software Engineering** edited by Charles Rich and Richard C. Waters, Morgan Kaufmann 1986.